



## **Algoritmos y estructuras de datos 1**

### **Proyecto Juego de Preguntas/Adivinanzas en Python**

**Jueves Noche - 2do Cuatrimestre 2025**

#### **Profesoras:**

- Monasterio, Maria Julia
- Giacin, Anabella

#### **Integrantes:**

- Naughton, Thomas
- Pagani Salem, Luca
- Godoy, Felipe
- Vazquez, Facundo

#### **Introducción**

El juego es una competencia interactiva de preguntas y adivinanzas para 2 a 4 jugadores, en el que los participantes se enfrentan en rondas eliminatorias. Cada jugador debe responder correctamente para mantenerse en juego; quienes fallen acumulan penalizaciones y eventualmente son eliminados.

El objetivo es que quede un solo ganador: el jugador que logre superar las rondas con mayor acierto gana.

#### **Dinámica de Juego**

- El juego inicia con todos los jugadores activos.
- En cada ronda se presenta una pregunta o adivinanza al jugador correspondiente.
- Responder correctamente mantiene al jugador en la partida y le otorga puntos.
- Responder incorrectamente resta puntos o acerca al jugador a la eliminación.
- A medida que los jugadores son eliminados, la dificultad de las preguntas y adivinanzas aumenta progresivamente, generando mayor reto para los que permanecen.
- Gana el último jugador en pie o, en caso de empate, el de mayor puntuación acumulada.

#### **Alcance del Producto**

#### **Beneficios**

- Diversión en grupo: Permite a 2–4 jugadores competir en un entorno amigable y entretenido.
- Aprendizaje y agilidad mental: el razonamiento lógico y el conocimiento general.
- Rejugabilidad alta: Gracias al banco de preguntas/adivanzas y el aumento progresivo de dificultad, cada partida es única.

## Objetivos

- Desarrollar un juego multijugador local (2 a 4 jugadores) en Python.
- Diseñar un sistema de rondas eliminatorias donde la dificultad aumenta conforme se reduce el número de jugadores.
- Incluir un banco inicial de preguntas y adivinanzas con categorías variadas y niveles de dificultad.
- Implementar un sistema de puntuación y eliminación claro por ronda y fase finalizada.

## Metas del Producto

- Funcionalidad básica multijugador local: Jugadores se turnan respondiendo desde la misma PC.
- Banco de contenido: Al menos 50 preguntas y 30 adivinanzas distribuidas en 4 rondas y 3 fases.
- Sistema de progresión de dificultad: Automáticamente incrementa la complejidad a medida que quedan menos jugadores o pasan a la siguiente fase/nivel.

## Resumen de funciones y características del software

- 1. Pantalla principal e inicio del juego**
  - Menú inicial con opciones: *Jugar*, *Instrucciones*, *Salir*.
  - Selección del número de jugadores (2–4).
  - Ingresar el nombre de cada jugador.
- 2. Gestión de preguntas y adivinanzas**
  - Banco de preguntas almacenado en un archivo ( .txt) o en el código.
  - Preguntas clasificadas en niveles de dificultad según la fase/nivel
  - Selección aleatoria de preguntas según el nivel actual.
- 3. Sistema de turnos multijugador**
  - Turnos secuenciales (jugador 1, luego jugador 2, luego jugador 3 y luego jugador 4) dependiendo de la cantidad seleccionada de jugadores.
  - El jugador responde seleccionando 1 de las respuestas que aparecen en pantalla.
  - Validación de respuesta: correcto / incorrecto.
- 4. Eliminación y progresión de dificultad**
  - Cada jugador tiene un número limitado de errores ( 3 vidas).
  - Si un jugador llega a 0 vidas, queda eliminado.
  - Cada vez que un jugador es eliminado, el nivel de dificultad sube para los que siguen.
- 5. Sistema de puntuación y ganador**
  - Correcto = +10 puntos.
  - Incorrecto = -5 puntos.
  - Mostrar la puntuación de todos los jugadores después de cada fase/nivel.
  - Gana el último jugador vivo o, si hay empate, el de mayor puntuación.
- 6. Sistema de puntuación y ganador**
  - Posibilidad de menú de opciones para finalizar partida con antelación dentro del turno.
  - Incorrecto = -5 puntos.
  - Mostrar la puntuación de todos los jugadores después de cada fase/nivel.
  - Gana el último jugador vivo o, si hay empate, el de mayor puntuación.

## Uso de Funciones, Bibliotecas y Técnicas

[random] -> Para elegir preguntas aleatorias.

[.txt] -> Para almacenar el banco de preguntas y adivinanzas.

Métodos:

**iniciar\_juego()**

enunciado, respuesta y nivel de dificultad. (Tendrá más métodos pequeños dentro)

**jugar\_ronda(), subir\_dificultad()**

controla rondas, turnos y avance de dificultad. (Tendrá más métodos pequeños dentro)

**verificar\_ganador(), mostrar\_puntuaciones()**

nombre, puntuación y vidas. (Tendrá más métodos pequeños dentro)

## Entregables

### Etapas 1 – Prototipo básico en consola (40%)

#### Diseño de Métodos:

- **Método 1** → con nombre, puntuación y vidas.
- **Método 2 y 3** → con enunciado, respuesta y nivel de dificultad (solo fáciles y medias en esta etapa).
- **Método 4 y 5** → para coordinar rondas y turnos entre los jugadores.

#### Carga de banco de preguntas/adivinanzas:

- Archivo .txt o directamente en el código con un mínimo de **15 preguntas/adivinanzas** de nivel fácil y medio.
- Lectura de preguntas en Python para mostrarlas en consola.

#### Sistema de turnos en consola:

- Pregunta secuencial a cada jugador.
- Validación de respuesta: correcto / incorrecto.
- Actualización de puntos y vidas.

#### Eliminación básica:

- Un jugador queda fuera al perder todas sus vidas.
- El juego continúa con los jugadores activos.

#### Determinación de ganador:

- Último jugador activo o mayor puntaje en caso de empate.

#### Pantalla de estado en consola:

- Mostrar puntuaciones y vidas al final de cada ronda.

## **Etapas 2 – Expansión de funcionalidades (80%)**

### **Banco de preguntas y adivinanzas ampliado:**

- Al menos 25 preguntas y 10 adivinanzas.  
Clasificadas en tres niveles de dificultad: fácil, medio y difícil.  
Selección aleatoria de preguntas según la fase y el nivel actual.

### **Sistema de progresión de dificultad:**

- A medida que los jugadores son eliminados, el nivel de dificultad aumenta automáticamente.  
Las fases se dividen en rondas con incremento progresivo de complejidad.

### **Sistema de puntuación mejorado:**

- Correcto = +10 puntos.  
Incorrecto = -5 puntos.  
Bonificación por rachas de respuestas correctas consecutivas.

### **Pantallas de estado mejoradas:**

- Mostrar puntuación, vidas y jugadores eliminados al final de cada fase.  
Resumen de desempeño general de todos los jugadores activos.

### **Menú ampliado:**

- Opción para ver instrucciones antes de iniciar la partida.  
Opción de reiniciar el juego sin salir del programa.

## **Etapas 3 – Refinamiento y extensiones (100%)**

### **Historial de partidas:**

- Registro en archivo .txt con: fecha, jugadores, puntajes finales, ganador y modo de juego.

### **Ranking global:**

- Archivo "ranking.txt" que almacena los mejores puntajes históricos.  
Visualización del top 5 jugadores desde el menú.

### **Nuevos modos de juego:**

- Clásico: vidas y eliminación (mecánica base).  
Contrarreloj: tiempo límite por respuesta.

### **Personalización de partida:**

- Selección de categorías (ejemplo: cultura general, lógica, adivinanzas).
- Configuración inicial de vidas y número de rondas.

**Logros y recompensas:**

- Racha perfecta: fase completada sin errores.
- Mejor remontada: jugador que estuvo al borde de la eliminación y terminó con puntaje alto.

**Menú principal completo:**

- Opciones: Jugar / Ranking / Instrucciones / Salir.

**Validaciones y robustez final:**

- Manejo de entradas inválidas sin que el juego se detenga.
- Mensajes motivacionales y descriptivos para mayor inmersión.